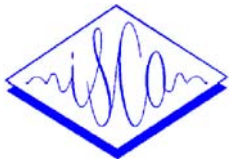

6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

A LANGUAGE-INDEPENDENT PROBABILISTIC MODEL FOR AUTOMATIC CONVERSION BETWEEN GRAPHEMIC AND PHONEMIC TRANSCRIPTION OF WORDS

Evangelos Dermatas and George Kokkinakis

WCL, Electrical Engineer and Computer Engineering Dept.
University of Patras, 26100 Patras, HELLAS.
e-mail:dermatas@george.wcl2.ee.upatras.gr

10.21437/Eurospeech.1999-459

ABSTRACT

In this paper we present a novel language-independent probabilistic model for automatic grapheme-to-phoneme and phoneme-to-grapheme conversion of words. In a fully unsupervised training procedure, two processes are applied; the transformation rules, which usually fail to provide the correct symbols, are eliminated, and new variable-length string transformation rules are defined improving the string transformation accuracy in the training data. In an iterative process the probabilistic transformation rules are updated in the direction of reducing the error rate of the transformed symbols. Long-term dependencies are defined automatically.

Training and testing of the model was carried out on lexicon and natural language corpora of six European Languages. Accurate generalisations have been achieved in all experiments for both transformation directions using a relative small number of defined rules in the training procedure. It is demonstrated that the variable-length probabilistic rules are sufficiently effective for describing bi-directional transcription.

1. INTRODUCTION

An important prerequisite for many tasks involving natural language processing is the automatic grapheme-to-phoneme (G2P) and phoneme-to-grapheme (P2G) conversion. Several transformation methods have been already proposed and tested [1-5,7-12]. The popular rule based systems such as DECtalk [1], MITalk [2], the NLR system [3], demonstrate impressive performance for some tasks. However, rule-based systems have an inherent problem concerning the extremely high cost and time consuming process of knowledge acquisition. Furthermore, additional effort is required in case of redefinition-modification-porting the transformation rules into new applications and languages.

A number of data-oriented and self-learning stochastic [12] and neural methods [4,7-9] have been proposed to face the cost of developing G2P and P2G conversion tools. In this case, induction methods, probabilistic models and neural network parameters are automatically estimated by applying learning algorithms in large databases of word-pronunciation pairs. The connectionist approach, in particular, is suitable for automatic extraction of generalisations but the input data must have constant size and the generalisation capability depends on the neural network architecture and its size. These disadvantages restrict the performance of the G2P conversion of the NETtalk to maximally 90% correct symbols [4]. The experiments in [7] show 66% phoneme accuracy and 26% word accuracy for the English language.

On the other hand, induction methods attempt to infer rules from training data. Table Look-up approaches are easy to implement, achieve fast transformation, and the well-defined hierarchical structure, the reduced tree (trie approach), can be established based on mutual information [3,8]. In the same framework the memory based learning methods [6,10] seem to achieve better generalisations when "forgetting" methods [11] are applied.

In this paper we propose a data-driven algorithm to produce probabilistic transformation rules based on lexical knowledge.

Specifically, in a database of string pairs an iterative algorithm retrieves and combines large chunk of sub-lexical elements. Initially each chunk is processed on the basis of an alignment between the equal-length lexical pairs using a Dynamic-Programming (DP) alignment algorithm. In an iterative process, variable length probabilistic rules are defined minimising the incorrectly transformed symbols. The proposed method can be used to produce multiple phonemic strings facing the problem of alternative pronunciations, as well as multiple graphemic forms for a specific phonetic string without referring to exception lists.

The data used for evaluating the proposed probabilistic string transformation model have been taken from the ESPRIT-261 project: 'Linguistic Analysis of the European Languages' and from our Lab's resources concerning Greek language data. A total number of approximately 80.000 grapheme-phoneme pairs of the Dutch, English, French, German, and Italian language have been extracted from a natural language text of approximately 1.8 million words. The Greek language lexical database consists of approximately 270000 string pairs.

The top-choice string accuracy of the model is greater than 60% for all languages and transformation directions without stress information. This performance compares favourably with those previously reported. Detailed results are presented below.

The structure of the paper is as follows. In the following two sections we present a detailed description of the proposed string conversion algorithm and the data-driven definition of variable-length probabilistic rules. In section 4 we present the corpora used for the evaluation. The experimental results are given in the next section for both conversion directions. We conclude this paper with a summary of the obtained results

.

2. THE PROBABILISTIC STRING TRANSFORMATION MODEL

The conversion G2P and P2G of words can be described as a string transformation problem. In general each N letter word is a string of N+2 graphemes:

$$G = (Sg_{st}, g_1, g_2 ,...,g_N, Sg_{end})$$

taken from the set of graphemic symbols $\Omega_G = \{ Sg_1 , Sg_2 ,..., Sg_G, Sg_{st}, Sg_{end} \}$. Two special symbols are added at the word boundaries, the word-starting symbol Sg_{st}, and the word-end symbol Sg_{end}. The corresponding phonemic transcription can be defined as:

$$P = (Sp_{st}, p_1, p_2 ,..., p_M, Sp_{end})$$

Each phonemic symbol belongs to the set of phonemes $\Omega_P = \{ Sp_{st}, Sp_1, Sp_2 ,..., Sp_P, Sp_{end} \}$. Any stress information is removed from both graphemic (only in the Greek language) and phonemic representation of words. The sets Ω_G, Ω_P are language dependent containing disjoint elements: $\Omega_P \cap \Omega_G = \{\}$. It is well known that in a great number of languages, exception rules of pronunciation are met at the word boundaries, e.g. in the French language. Therefore string boundary symbols are used to define transformation rules applicable only in the prefix and suffix part of the transformed string.

In the rest of this paper we discuss only the G2P problem. The same method is applied also for the P2G transcription by interchanging the role of symbol strings.

2.1 The string transformation model

Let a set of C probabilistic sub-string transformation rules:

$$\Omega_C = \{ r_1, r_2, \ldots, r_C \}$$

where $r_{i=}(Gi \rightarrow Pi, Prob(Gi \rightarrow Pi)), i=1,C$.

Initialization ($t \leftarrow 0$).

The graphemic sequence of symbols is defined as the target string:

$$T(0)' \leftarrow (Sg_{st}, g_1, g_2 ,...,g_N, Sg_{end})$$

The following iterative process transforms a sub-string of graphemic symbols into a sequence of phonemic symbols.

Sub-string transformation ($t \leftarrow t+1$).

From the set of all applicable transformation rules, the most rich in context are selected:

$$Tr(t) = \{ r_m : Gm \in T(t-1)' \wedge |Gm| = \max(|Gj|), Gj: \in T(t-1)' \}$$

where the symbol string, T(t-1)' contains, in general, both graphemic and phonemic symbols,

$$T(t-1)' = (s_1, s_2, \ldots, s_{N(t-1)})$$

|s| is the number of symbols in the string s, N(t-1) is the length of the string T(t-1)', and max(|Gj|) is the maximum length of all applicable rules.

The most probable rule(s) r_i in the set Tr(t):

$$r_i = \{ r_m : Prob(r_m) = \max(Prob(r_j)), \ r_j \in Tr(t) \}$$

is used to transform part of the string T(t-1)':

$$T(t)' \leftarrow T(t-1)' \Leftrightarrow$$
$$(s_1,s_2, \ldots,s_k,Pi,\ldots,s_{N(t)}) \leftarrow (s_1,s_2,\ldots,s_k,Gi,\ldots,s_{N(t-1)})$$

k is the position where the rule r_i is applied.

In case where the rule r_i is applicable in multiple positions the nearest to the string end is selected.

Termination condition.

The sub-string transformation process is terminated in case of exhaustion of the applicable rules: Tr(t) = {}.

The proposed algorithm cannot guarantee the transformation of all graphemic symbols. This condition can be satisfied in the case where any individual graphemic symbol in the rules set can be transformed to a phonemic symbol(s).

The basic idea behind the proposed transformation algorithm is that an unknown word should be converted on the basis of lexical knowledge where the exceptions (the longer transformation rules) are preferred. In addition we avoid the commonly used left-to-right transformation scheme which leads to cumulative transformation errors.

3. THE TRAINING ALGORITHM

The rules definition process is fully automatic and language independent. The proposed algorithm does not require syllabification boundaries or segmental markers.

Given the training data, a pair set of graphemic and phonemic strings,

$$\Omega_T = \{(Gi, Pi), i=1,M\},$$

the following data-driven algorithm establishes variable-length string probabilistic rules:

Definition of one-symbol probabilistic rules ($t \leftarrow 0$).

An accurate definition of one-symbol transformation rules requires alignment of the training data. In case that this information is not available explicitly, the alignment problem can be faced using dynamic programming algorithms.

The well-known Viterbi algorithm has already been employed to locate the most likely position of each phonemic symbol relative to the graphemes, but extended experiments showed that the optimisation criterion leads to systematically wrong alignment. In a second approach pairs of grapheme and phoneme strings of equal length are used to initialise the set of transformation rules and to define the corresponding probabilities. Experiments showed that the latter approach gives comparable results to the Viterbi algorithm.

In our work the second approach is used to define the initial one-symbol transformation rules by assuming one-to-one symbol alignment. The selection of equal length strings reduces the errors arising from the hypothesis of one-to-one symbol alignment. All rule-probabilities are estimated using the rule frequency of occurrence in the sub-set of equal length training data. The most probable rule for each graphemic symbol is transferred into the initial set of probabilistic rules.

Generally, in the initialisation process, the following set of probabilistic rules is defined:

$$\Omega_C(t) = \{ r_1, r_2, \ldots r_i, \ldots, r_{C(t)} \}$$

where |Gi| = |Pi| = 1, C(t) = N,

$$Prob(Gi \rightarrow Pi) = \frac{N(Gi \rightarrow Pi)}{\sum_{Pi} N(Gi \rightarrow Pi)}$$

and $N(Gi \rightarrow Pi)$ is the number of times the rule r_i is applied in the process of transforming the equal length string from the graphemic to phonemic form.

The following two processes update the initial set of rules recursively.

Removing the frequently failing rules (t←t+1).
The proposed probabilistic string transformation model Pstm(.) is used to provide phonemic sequences for each graphemic sequence of the training data. This deterministic process gives a unique phonemic sequence:

$$Pi' = Pstm(Gi)$$

Then, appropriate alignment path between Pi' and Pi is detected using a dynamic-programming technique. This is based on the hypothesis that, if the optimum path cost in positions $(p_1\text{-}1,p_2\text{-}1)$, $(p_1,p_2\text{-}1)$, $(p_1\text{-}1,p_2)$, is known, the path cost in the position (p_1,p_2) can be estimated by the following recurrent equation:

$$Co(p_1,p_2) = \min \begin{cases} Co(p_1\text{-}1,p_2) + Cnt \\ Co(p_1\text{-}1,p_2\text{-}1) + Cost(Pi'(p_1),Pi(p_2)) \\ Co(p_1,p_2\text{-}1) + Cnt \end{cases}$$

where, $Co(p_1,p_2)$ is the cost of reaching the alignment, $Cost(Pi'(p_1), Pi(p_2))$ is set to 1 when the p_1 symbol of the string Pi' is not identical to the symbol in position p_2 of the string Pi, otherwise is set to 0.

A small positive value is set for the constant cost *Cnt*. The optimum *Cnt* value is detected experimentally. In our

$$I(p_1(m),p_2(m)) = \arg\min_{p_1,p_2} \begin{cases} Co(p_1(m\text{-}1)\text{-}1,p_2(m\text{-}1)) \\ Co(p_1(m\text{-}1)\text{-}1,p_2(m\text{-}1)\text{-}1) \\ Co(p_1(m\text{-}1),p_2(m\text{-}1)\text{-}1) \end{cases}$$

experiments it was set to 1.

The backtracking recurrent equation leads to the estimation of the optimum alignment path:

Initial value: $p_1(0) = |Pi'|$, $p_2(0) = |Pi|$).

Based on the alignment path information, the frequency of creating correct phonemic symbols for each rule is estimated in the set of training data. A rule creates correct phonetic symbols only if the phonetic sequence of the rule in the alignment path is mapping unique identical symbols.

The probability of detecting the correct phonemic sequence is used to update each rule's probability:

$$Prob(Gi \rightarrow Pi) = \frac{Nc(r_i)}{N(r_i)}$$

where, $Nc(r_i)$ is the frequency of applying successfully the rule r_i and $Nc(r_i)$ is the total number of rule usage in the training set.

The less effective rules are eliminated from the set of active rules:

$$\Omega_{C(t)} = \Omega_{C(t-1)} - \{ r_i : Prob(Gi \rightarrow Pi) < Th(t) \}$$

A number of alternative methods can be employed to define the probability threshold Th(t). In our approach the rule elimination threshold was set to the probability of detecting the correct phoneme symbol.

Definition of new rules.
The remaining rules $\Omega_{C(t)}$ are used to re-transform the training data. In the failed positions, longer (in graphemic symbols and probably phonemic symbols) rules are created. Let us assume that the rule r_i failed to provide the correct phonemic sequence in a training example. The phonemic symbol boundaries where the alignment information gave correct symbol detection define the new rule phonemic chunk. The new graphemic chunk is simply the grapheme sequence that produces the phonemic chunk, enlarged by the boundary grapheme at the beginning or at the chunk end, randomly selected.

Termination condition.
The convergence of the iterative procedure is not guaranteed. Generally, during the rules adaptation process, the probability threshold Th(t) increases. Consequently the number of new rules and the symbol error rate decreases. Unfortunately, no simple criteria exist which can be used for determining the end of the rule definition method. One possible termination criterion is to compare two successive probabilities of detecting the correct phonemic symbol. When no significant changes take place any longer the re-estimation process is terminated. The experiments verified that the best results are obtained after just two or three iterations.

4. THE CORPUS DATABASE

The database used contains texts of six European languages and the corresponding lexicons, which have been manually or semi-automatically converted into phonetic representation in the framework of the ESPRIT-291/860 project "Linguistic Analysis of the European Languages". The database does not include alignment information of the grapheme and phoneme pairs, therefore experiments on the alignment capabilities of the proposed transformation method cannot be carried out.

In table 1 the corpus and the lexicon size for each language is given followed by the mean word (grapheme and phoneme string) length. In table 2 the number of grapheme and phoneme symbols and the estimated symbol perplexity both in corpus and lexicon are given.

Table 1 Corpus and lexicon size, and mean word length.

Lang.	Corpus Size	Lexicon Size	Mean word length (grapheme/ phoneme)	
			Corpus	Lexicon
Dutch	291108	29607	5.650/5.169	9.529/8.712
English	327422	29540	5.028/4.523	7.634/6.973
French	259618	26244	5.229/4.142	8.273/6.565
Germ.	175426	15242	5.142/5.796	9.389/9.844
Greek	--------	279070	-------	10.249/9.79
Italian	295438	31357	5.445/5.282	8.291/8.137

Table 2 Number of grapheme symbols and string perplexity in lexicon and corpus.

Language	Number of Symbols	Lexicon perplexity	Corpus Perplexity
Dutch	48	7.559	7.335
English	51	7.520	7.451
French	64	7.572	7.427
German	58	7.279	7.046
Greek	36	9.046	9.046
Italian	38	6.761	6.831

Table 3 Number of phoneme symbols and string perplexity in lexicon and corpus.

Language	Number of Symbols	Lexicon perplexity	Corpus perplexity
Dutch	**49**	**9.831**	**9.652**
English	**39**	**9.842**	**10.086**
French	**32**	**8.732**	**8.667**
German	**45**	**9.468**	**9.295**
Greek	**30**	**6.858**	**6.858**
Italian	**30**	**7.477**	**7.447**

5. EXPERIMENTAL RESULTS

Each language corpus and lexicon was split into two sets. The training set was four times greater than the testing set of string pairs. In the following tables the word and symbol accuracy of the G2P and the P2G transcription is shown.

Table 4 Percent word and symbol accuracy of the G2P conversion measured on lexicon and corpus data

Language	Words Lexicon/corpus	Symbols Lexicon/corpus
Dutch	**75.43/77.91**	**80.55/84.98**
English	**63.73/68.38**	**85.22/85.94**
French	**67.48/68.11**	**83.46/84.98**
German	**72.15/78.54**	**79.34/81.77**
Greek	**93.55/94.34**	**95.23/97.34**
Italian	**90.11/92.99**	**94.34/95.56**

Table 5 Percent word and symbol accuracy of the P2G conversion measured on lexicon and corpus data.

Language	Words Lexicon/corpus	Symbols Lexicon/corpus
Dutch	**76.55/78.75**	**83.43/86.55**
English	**61.44/67.71**	**81.87/83.54**
French	**65.34/69.62**	**82.49/84.22**
German	**70.35/74.77**	**79.34/81.77**
Greek	**Currently tested**	
Italian	**89.28/92.65**	**93.31/96.55**

6. CONCLUSION

A language-independent probabilistic model for automatic G2P and P2G transcription of words has been formulated and evaluated on the ESPRIT-291/860 lexicon and corpus database of six European languages. The most important advantages of the proposed model are its automatic data-driven learning process, the definition of variable-length rules where exceptions can be stored, and the hierarchy in the string transformation process: the longer and most reliable rules are first applied. The experimental results have shown exceptional behaviour of the model both in the training process, where long-term graphemic and phonemic chunks are successfully detected and in the string transformation process where accurate conversion of the desired target string is achieved.

REFERENCES

[1] Allen, J., Hunnicutt, S., and Klatt, D. "From Text to Speech: The MITalk System", Cambridge, UK: Cambidge University Press.1987

[2] Andersen, R Kuhn, A Lazarides, P Dalsgaard, J Haas, E Noth (1996). 'Comparison of Two Tree-Structured Approaches for Grapheme-to-Phoneme Conversion'. Proceedings of International Conference on Spoken Language Processing, Philadelphia, Oct., pp

[3] Andersen, O., and Dalsgaard, P., "Multi-lingual testing of a self-learning approach to phonemic transcription of orthography". In *EUROSPEECH95*, pages 1117–1120, September 1995.

[4] Sejnowski, T.J., and Rosenberg, C.R., "Parallel networks that learn to pronounce English text". *Complex Systems*, pages 145–168, 1987.

[5] W. Daelemans, A. Bosch, "Language-Independent Data-Oriented Grapheme-to-Phoneme Conversion", Van Santen, J., R. Sproat, J. Olive, and J. Hirschberg (eds.) *Progress in Speech Synthesis*. New York: Springer Verlag, 77-90, 1996.

[6] Quinlan, J. 1993. C4.5: Programs for Machine Learning. San Mateo, CA: Morgan Kaufmann.

[7] Lucas S. M., Damper R. I., "Syntactic neural networks for bi-directional text-phonetics translation", In G. Bailly and C Benoit (ed), Talking Machines, Theories, Models and Designs, North-Holland , 1992.

[8] Dalsgaard P., Andersen O., Hansen A., "Theory and Application of two Approaches to Grapheme-to-Phoneme Conversion", Deliverable 4.6, LRE-ONOMASTICA Project, CPK,1995.

[9] Wolters M., "A Dual Route Neural Net Approach to Grapheme-to-Phoneme Conversion", Lecture notes COMP-SCI , vol. 1112, pp. 233-239, 1996.

[10] Deligne S., Yvon F., Bimbot F., "Variable-length sequence matching for phonetic transcription using joint multigrams", EUROSPEECH95, pp. 2243-2246.

[11] Daelemans W., Bosch A., Zavrel J., "Forgetting Exceptions is Harmful in Language Learning", Kluwer Academic Pulishers.

[12] Rentzepopoulos P., Kokkinakis G., "Efficient multilingual Phoneme-to-Grapheme conversion based on HMM", Computational Linguistics, Vol. 22, No. 3, pp. 319-376, 1996.